Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) An article comprising a machine-readable medium embodying instructions that when performed by one or more machines result in operations comprising:

determining whether a storage device, in a data processing system running an operating system, includes ~~a protected area~~ storage area protection, the operating system including a hardware abstraction layer;

removing the storage area protection of the storage device from within the running operating system and without rebooting the data processing system, thereby creating a formerly protected storage area; and

providing information derived from the formerly protected storage area to a data processing system detection tool;

wherein said determining and said removing occur in a kernel mode of the data processing system.

2. (Original) The article of claim 1, wherein the operating system further includes a graphical user interface (GUI), virtual memory management and multitasking.

3. (Currently Amended) The article of claim 1, wherein determining whether the storage device includes the ~~protected area~~ storage area protection comprises:

checking whether the storage device supports a protected area specification; and

identifying a protected storage capacity and an unprotected storage capacity of the storage device.

4. (Original) The article of claim 1, wherein removing the storage area protection comprises volatilely resetting a storage address value.

5. (Original) The article of claim 4, wherein resetting a storage address value comprises calling a MAX ADDRESS command.

6. (Cancelled)

7. (Original) The article of claim 4, wherein the storage area protection of the storage device is restored by the data processing system upon system reboot, leaving the storage device unaltered.

8. (Original) The article of claim 1, wherein the operations further comprise:
scanning the formerly protected storage area; and
identifying file system information in the formerly protected storage area.

9. (Original) The article of claim 1, wherein providing the information derived from the formerly protected storage area comprises sending the information over a transport medium to the data processing system detection tool.

10. (Original) The article of claim 9, wherein the operations further comprise reconstructing a file system of the formerly protected storage area to derive the information.

11. (Original) The article of claim 9, wherein providing the information derived from the formerly protected storage area further comprises selecting the transport medium from a group including a peripheral device interface medium and a network communications medium.

12. (Currently Amended) An article comprising a machine-readable medium embodying instructions that when performed by one or more machines result in operations comprising:

determining whether a storage device, in a data processing system running an operating system, includes ~~a protected area~~ storage area protection, the operating system including a hardware abstraction layer;

removing the storage area protection of the storage device from within the running operating system and without rebooting the data processing system, thereby creating a formerly protected storage area; and

providing information derived from the formerly protected storage area to a data processing system detection tool;

wherein providing the information derived from the formerly protected storage area comprises sending the information over a transport medium to the data processing system detection tool;

wherein providing the information derived from the formerly protected storage area further comprises selecting the transport medium from a group including a peripheral device interface medium and a network communications medium; and

wherein sending the information over the transport medium comprises sending the information in packets having a packet structure useable over both the peripheral device interface medium and the network communications medium.

13. (Original) The article of claim 12, wherein the packet structure is useable over a Universal Serial Bus (USB) and over an Internet Protocol (IP) network.

14. (Original) The article of claim 12, wherein the packet structure includes a packet identifier field, and the operations further comprise specifying a detection-tool packet identifier for each packet.

15. (Original) The article of claim 12, wherein the packet structure allows for only a one-to-one connection.

16. (Original) The article of claim 12, wherein the packet structure specifies small packets to reduce latency.

17. (Currently Amended) A method comprising:

loading a kernel-mode software module in a computing system running an operating system; and

without rebooting the computing system, using the kernel-mode software module to perform operations from within the operating system, the operations comprising

determining whether a storage device in the computing system includes ~~a protected area~~ storage area protection, and

reversibly removing the storage area protection, thereby creating a formerly protected storage area.

18. (Original) The method of claim 17, wherein loading the kernel-mode software module comprises communicatively coupling a machine-readable medium with the computing system, a detection agent being tangibly embodied in the machine-readable medium to run and dynamically load the kernel-mode software module without altering the storage device.

19. (Original) The method of claim 18, wherein the machine-readable medium comprises an optical disk.

20. (Original) The method of claim 17, further comprising:

scanning the formerly protected storage area; and

identifying file system information in the formerly protected storage area.

21. (Original) The method of claim 17, further comprising sending information derived from the formerly protected storage area over a selected transport medium to a data processing system detection tool.

22. (Original) The method of claim 21, wherein sending the information over the selected transport medium comprises sending the information in packets having a packet structure useable over both a peripheral device interface medium and a network communications medium.

23. (Original) The method of claim 22, wherein the packet structure includes a packet identifier field used by the detection tool, and the packet structure specifies small packets to reduce latency.

24. (Original) A system comprising:

a data processing system detection tool; and

a kernel-mode software module operable to provide the detection tool with access to a protected area of a storage device in a data processing system when the kernel-mode software module is loaded into the data processing system.

25. (Original) The system of claim 24, wherein the detection tool is operable from within the data processing system to access the storage device over a bus, the system further comprising a hardware write blocker operable to allow the kernel-mode software module access to a firmware command.

26. (Original) The system of claim 24, wherein the detection tool is operable as a stand alone application and as a client application.

27. (Original) The system of claim 24, further comprising a detection agent operable to send information to the detection tool, the detection agent being operable to load the kernel-mode software module in the data processing system and communicate with the loaded kernel-mode software module and with the detection tool.

28. (Original) The system of claim 27, wherein the detection agent is further operable to reconstruct a file system of the protected storage area and send the reconstructed file system information to the detection tool.

29. (Original) The system of claim 27, wherein the detection agent is further operable to select a transport medium from a group including a peripheral device interface medium and a network communications medium, and the detection agent communicates with the detection tool using a common a packet structure useable over both the peripheral device interface medium and the network communications medium.

30. (Original) The system of claim 29, wherein the packet structure includes a packet identifier field used by the detection tool, and the packet structure specifies small packets to reduce latency.

31. (Original) The system of claim 24, further comprising a software write blocker.

32. (Original) The system of claim 24, wherein the detection tool comprises a computer forensics tool.

33. (Original) The system of claim 24, wherein the kernel-mode software module comprises a device driver.

34. (Original) The system of claim 33, wherein the device driver comprises a Windows Driver Model (WDM) driver.

35. (Original) The system of claim 33, wherein the storage device comprises an ATA hard disk.

36. (Currently Amended) A system comprising:

means for directly accessing a protected area of a storage device in a data processing system live from a high level operating system without a reboot; and

means for delivering information derived from the protected storage area to a data processing system detection tool;

wherein the means for delivering comprises multi-transport means for delivering the information, including means for selectively communicating over a network communications medium or a peripheral device interface medium to support remote imaging and analysis of the directly accessed protected area.

37. (Cancelled)